



Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

**Sistemi Operativi 1 — A.A. 2004-2005, prova scritta del 26 aprile 2005**

## Esercizio 4

1. Mostra lo schema di traduzione di indirizzo virtuale in indirizzo fisico per una architettura di memoria virtuale con paginazione e segmentazione. Mostra anche la struttura della tabella delle pagine e dei segmenti.

2. Considera l'architettura al punto 1 con un indirizzo di memoria virtuale di 32 bit e pagina di 4Kbytes. L'istruzione macchina "copy from 0x00001ffe to 0x00002ffe" occupa 9 bytes a partire dall'indirizzo 0x00000ffe e copia una parola di 4 bytes. Dati e istruzioni sono tutti nello stesso segmento e tutte le tabelle sono residenti. Quanti page faults può generare al più l'esecuzione di questa sola istruzione? Spiega.

## Esercizio 5

Considera una politica di scheduling feedback con 10 livelli di priorità  $p$  (0=alta priorità, 9=bassa priorità) in cui si pone  $p=p+1$  ogni volta che scade il quanto  $q$  del processo (eventualmente saturando a 9). Ogni volta che il processo va in blocco ritorna in una delle code ready con  $p=p-2$  (eventualmente saturando a 0). Il rilascio anticipato della cpu dovuto ad una system call bloccante non dà luogo ad alcun credito di quanto.

1. Supponi che un processo non vada mai in blocco, quale sarà l'evoluzione del valore di  $p$  per tale processo?

2. Supponi che un processo vada in blocco ogni  $6,5q$ , quale sarà l'evoluzione del valore di  $p$  per tale processo?



## Esercizio 4

1. Mostra lo schema di traduzione di indirizzo virtuale in indirizzo fisico per una architettura di memoria virtuale con paginazione a due livelli. Mostra anche la struttura delle tabelle delle pagine.

2. Considera l'architettura al punto 1 con un indirizzo di memoria virtuale di 32 bit e pagina di 4Kbytes. Assumi che tutte le tabelle delle pagine siano sempre residenti. L'istruzione macchina "call 0x00050000" occupa 5 bytes a partire dall'indirizzo 0x00000ffe ed è una chiamata di procedura. Lo stack pointer contiene il valore 0x80001001 che punta alla prima locazione libera dello stack. Lo stack cresce verso indirizzi di memoria bassi (cioè lo stack pointer viene decrementato a seguito di una operazione di push). Quanti page faults può generare l'esecuzione di tale istruzione? Spiega.

## Esercizio 5

Considera una politica di scheduling feedback con 7 livelli di priorità  $p$  (0=alta priorità, 6=bassa priorità) in cui si pone  $p=p+2$  ogni volta che scade il quanto  $q$  del processo (eventualmente saturando a 6). Ogni volta che il processo va in blocco ritorna in una delle code ready con  $p=p-5$  (eventualmente saturando a 0). Il rilascio anticipato della cpu dovuto ad una system call bloccante non dà luogo ad alcun credito di quanto.

1. Supponi che un processo vada sempre in blocco dopo  $1,5q$ , quale sarà l'evoluzione del valore di  $p$  per tale processo?

2. Supponi che un processo vada in blocco ogni  $6,5q$ , quale sarà l'evoluzione del valore di  $p$  per tale processo?